

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

As rescanning documents *will not* correct images,  
Please do not report the images to the  
Image Problem Mailbox.

**THIS PAGE BLANK (USPTO)**

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-207202

(43)Date of publication of application : 28.07.2000

(51)Int.CI. G06F 9/22  
G06F 9/28

(21)Application number : 11-307684 (71)Applicant : PACIFIC DESIGN KK

(22)Date of filing : 28.10.1999 (72)Inventor : SATO TOMOMI

## (30)Priority

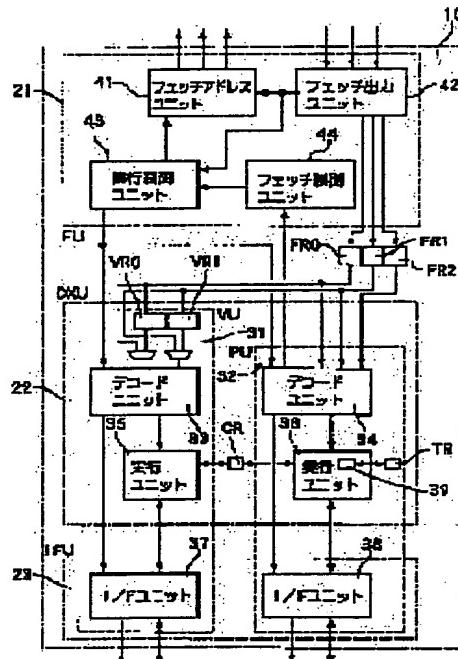
Priority number : 10308673 Priority date : 29.10.1998 Priority country : JP  
99 287565 06.04.1999 US

## (54) CONTROLLER AND DATA PROCESSOR

### (57)Abstract:

PROBLEM TO BE SOLVED: To provide the controller that can perform a process, which needs to be performed in real time, at a high speed and flexibly cope with alterations, extensions or the like.

SOLUTION: A microcode controller 10 of a microcode control system is provided with a dedicated data processing unit (VU) 31 which can perform a process using a dedicated circuit, a general data processing unit(PU) 32, and a fetch unit 21 which is common to them. This controller 10 can synchronously control the dedicated data processing unit 31 and the general data processing unit 32 with dedicated instructions and general instructions described in microcode on an instruction code level, so that multiprocessing can be performed by these units, without increasing the circuit scale. Consequently, the controller for high-speed processing that has high cost performance which includes the development cost or the like and is suitable for processing which is demanded to be processed in real time and the data processor which uses it can be provided.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

**THIS PAGE BLANK (USPTO)**

(19)日本国特許庁 (JP)

## (12) 公開特許公報 (A)

(11)特許出願公開番号

特開2000-207202

(P2000-207202A)

(43)公開日 平成12年7月28日(2000.7.28)

(51) Int.Cl. <sup>7</sup> G 0 6 F	識別記号 9/22 3 2 0	F I G 0 6 F 9/22	テマコード(参考) 3 2 0 Z
9/28	3 2 0	9/28	3 2 0 D 3 2 0

審査請求 未請求 請求項の数20 O L (全16頁)

(21)出願番号	特願平11-307684
(22)出願日	平成11年10月28日(1999.10.28)
(31)優先権主張番号	09/287565
(32)優先日	平成11年4月6日(1999.4.6)
(33)優先権主張国	米国(US)
(31)優先権主張番号	特願平10-308673
(32)優先日	平成10年10月29日(1998.10.29)
(33)優先権主張国	日本(JP)

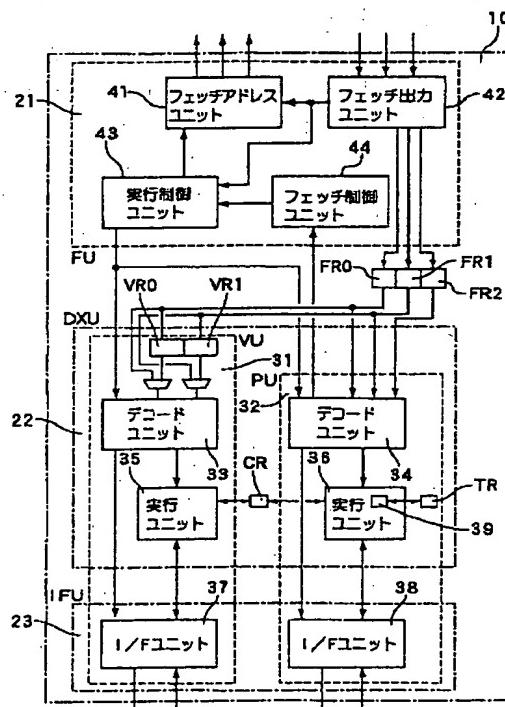
(71)出願人	598149242 パシフィック・デザイン株式会社 東京都新宿区西新宿6丁目12番1号
(72)発明者	佐藤友美 茨城県つくば市東2丁目18番地10ルーム つくば31号202
(74)代理人	100102934 弁理士今井彰

## (54)【発明の名称】制御装置およびデータ処理装置

## (57)【要約】

【課題】 リアルタイム性が要求される処理を高速で実行可能であり、さらに、変更あるいは拡張などにフレキシブルに対応できる制御装置を提供する。

【解決手段】 専用回路を用いた処理を実行可能な専用データ処理ユニット(VU)31と、汎用データ処理ユニット(PU)32と、これらに対し共通のフェッчуユニット21を備えたマイクロコード制御方式の制御装置10を提供する。この制御装置10では、マイクロコードで記述された専用命令および汎用命令により命令コードレベルで専用データ処理ユニット31と汎用データ処理ユニット32を同期制御することが可能であり、回路規模を大きくすることなく、それらのユニットにおいて多重処理を行うことができる。このため、開発費用などを含めたコストパフォーマンスが高く、リアルタイム性が要求される処理に適した高速処理用の制御装置およびこれを用いたデータ処理装置を提供できる。



## 【特許請求の範囲】

【請求項 1】 特定のデータ処理に適した専用回路に対する専用命令をデコードおよび実行可能な少なくとも 1 つの専用データ処理ユニットと、

前記専用命令以外の処理を行う汎用命令をデコードおよび実行可能な汎用データ処理ユニットと、

マイクロコード化された前記専用命令および汎用命令を記憶したコードメモリから命令コードをフェッチし、前記専用データ処理ユニットおよび汎用データ処理ユニットに供給するフェッチユニットとを有する制御装置。

【請求項 2】 請求項 1において、前記命令コードは、前記専用命令および汎用命令、さらに複数の前記専用データ処理ユニットを有するときはいずれかの専用データ処理ユニットの専用命令であることを識別可能な識別コードを備えており、

前記フェッチユニットは、前記識別コードをデコードして前記汎用データ処理ユニットまたはいずれかの専用データ処理ユニットに対しデコードおよび実行を指示する実行制御部を備えていることを特徴とする制御装置。

【請求項 3】 請求項 1において、前記命令コードは、後続の命令コードが前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかにおいて並列処理可能であることを示す並列処理フラグを備えており、さらに、

前記フェッチユニットは、前記命令コードの並列処理フラグに基づき前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかに対し前記後続の命令コードのデコードおよび実行を指示する実行制御部とを備えている制御装置。

【請求項 4】 請求項 1において、さらに、少なくとも前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかのための実行レジスタを有し、さらに、前記フェッチユニットは、前記実行レジスタに前記命令コードをストアすると同時に後続の命令コードをフェッチするフェッチデータ出力部を備えている制御装置。

【請求項 5】 請求項 4において、各々の前記専用データ処理ユニットのための前記実行レジスタを有する制御装置。

【請求項 6】 請求項 1において、前記命令コードは、後続の命令コードが前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかにおいて並列処理可能であることを示す並列処理フラグを備えており、さらに、

前記フェッチユニットは、少なくとも前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに

10

複数の前記専用データ処理ユニットを有するときはそれらのいずれかのための実行レジスタに前記命令コードをストアすると同時に前記後続の命令コードをフェッチするフェッチデータ出力部と、

前記命令コードの並列処理フラグに基づき前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかに対し前記後続の命令コードのデコードおよび実行を指示する実行制御部とを備えている制御装置。

【請求項 7】 請求項 1において、前記命令コードは、該命令コードのワード長を示す命令長コードを備えており、さらに、

前記フェッチユニットは、前記命令コードに続く最大命令長分のマイクロコードを後続の命令コードとしてフェッチするフェッチデータ出力部を備えている制御装置。

【請求項 8】 請求項 7において、前記フェッチユニットは、さらに、2 ライン分の命令フェッチアドレスを出力するフェッチアドレス出力部を備えている制御装置。

【請求項 9】 請求項 1において、前記フェッチユニットは、命令フェッチアドレスを出力するフェッチアドレス出力部と、

このフェッチアドレス出力部に対しテンポラルレジスタに格納された復帰アドレスを供給可能なフェッチ制御部とを備えており、さらに、

汎用データ処理ユニットは、分岐または割込み命令を実行するときに復帰アドレスをメモリに記憶すると共に前記テンポラルレジスタにストアし、復帰処理を実行するときに次の復帰アドレスを前記メモリから前記テンポラルレジスタにロードするレジスタ管理機能を備えている制御装置。

【請求項 10】 請求項 9において、前記レジスタ管理機能は、割込み復帰アドレス用およびコンディションコード用の前記テンポラルレジスタを各々備えており、割込み命令を実行するときに復帰アドレスおよびコンディションコードを前記メモリに記憶すると共に、前記割込み復帰アドレス用および前記コンディションコード用のテンポラルレジスタに各々ストアし、復帰処理を実行するときに復帰アドレスのメモリアドレスおよびコンディションコードのメモリアドレスを出力し次の復帰アドレスおよびコンディションコードを前記メモリから前記テンポラルレジスタに各々ロードすることを特徴とする制御装置。

【請求項 11】 請求項 1ないし 10 のいずれかに記載の制御装置と、この制御装置によって制御可能な少なくとも 1 つの前記専用回路とを有するデータ処理装置。

【請求項 12】 請求項 11において、前記専用回路は、符号化されたデータストリームからスタートコードを検出する回路、または、可変長符号化されたデータを連続的に復号する回路であることを特徴とするデータ処理装置。

50

理装置。

【請求項13】 請求項1に記載の制御装置の命令コードであって、前記専用命令および汎用命令、さらに複数の前記専用データ処理ユニットを有するときはいずれかの専用データ処理ユニットの専用命令であることを識別可能な識別コードを備えている前記命令コードが記録されていることを特徴とする記録媒体。

【請求項14】 請求項13において、後続の命令コードが前記専用データ処理ユニットのいずれか、または前記汎用データ処理ユニットにおいて並列処理可能であることを示す並列処理フラグをさらに備えた前記命令コードが記録されていることを特徴とする記録媒体。

【請求項15】 請求項13または14において、当該命令コードのワード長を示す命令長コードを備えた前記命令コードが記録されていることを特徴とする記録媒体。

【請求項16】 請求項1に記載の制御装置の制御方法であって、前記命令コードは、前記専用命令および汎用命令、さらに複数の前記専用データ処理ユニットを有するときはいずれかの専用データ処理ユニットの専用命令であることを識別可能な識別コードを備えており、前記識別コードをデコードして前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかに対しデコードおよび実行を指示する工程を有する制御方法。

【請求項17】 請求項16において、前記命令コードは、後続の命令コードが前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかにおいて並列処理可能であることを示す並列処理フラグを備えており、さらに、前記命令コードの並列処理フラグに基づき前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかに対し前記後続の命令コードのデコードおよび実行することを指示する工程をさらに有する制御方法。

【請求項18】 請求項17において、さらに、前記制御装置は、少なくとも前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかのための実行レジスタを有しており、前記実行レジスタに前記命令コードをストアすると同時に後続の命令コードをフェッチする工程と、前記命令コードの並列処理フラグに基づき前記汎用データ処理ユニットまたは前記専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかに対し前記後続の命令コードのデコードおよび実行を指示する工程とをさらに有する制御方

法。

【請求項19】 請求項16において、前記命令コードは、該命令コードのワード長を示す命令長コードを備えており、さらに、

前記命令コードに続く最大命令長分のマイクロコードを後続の命令コードとしてフェッチする工程をさらに有する制御方法。

【請求項20】 請求項16において、テンポラルレジスタに格納された命令フェッチアドレスを出力する工程と、

分岐または割込み命令を実行するときに復帰アドレスをメモリに記憶すると共に前記テンポラルレジスタにストアする工程と、

復帰処理を実行するときに次の復帰アドレスを前記メモリから前記テンポラルレジスタにロードする工程とをさらに有する制御方法。

【発明の詳細な説明】

#### 【0001】

【発明の属する技術分野】 本発明は、マイクロコードにより制御される制御装置に関し、特に、リアルタイム性の要求される通信装置や復号装置・符号化装置などのデータ処理装置に適した制御装置、それに適した命令コードおよび制御方法に関するものである。

#### 【0002】

【従来の技術】 従来、リアルタイム性を強く要求される画像データ、たとえば、MPEG1（シーンチェンジの多いゲーム用途などの場合）あるいはMPEG2などで規格化されたデータストリームの復号・符号化装置などの高速処理が要求されるシステムは専用回路（ハードウェイアあるいはワイヤードロジック）で実現されることが多い。

【0003】 これに対し、汎用的なデータ処理装置としては、一般的にマイクロプロセッサー（MPU）やデジタル・シグナルプロセッサー（DSP）を用いたものが多くの用いられているが、このような演算機能を内蔵した規模のプロセッサは、1クロックを問題とするような利用には適していない。特に高速処理が要求される分野ではMPUあるいはDSPでも高速動作保証のある高価なMPUやDSPを採用すると共に、それらを高い周波数で動作させながら使用することが必要となる。さらに、システムレベルあるいはアプリケーションレベルでも、ある程度タイミング・クリティカルバスが発生しないように時間的許容度を持たせる必要がある。したがって、MPUあるいはDSPを搭載した高速のデータ処理装置を実現するには、実装面でコスト的にも技術的にも無理な点が多い。

【0004】 近年、半導体のプロセス技術の進歩により、高速動作可能なMPUやDSPが利用可能になるに連れてこれらの適用範囲は大きく広がっている。しかしながら、上記のように、基本動作周波数が高く1クロッ

クで一つの処理を確実に実行することが要求される復号・符号化処理のような分野へはそれほど多く適用されていない。

【0005】このため、基本動作周波数がある程度限られた高速性の要求されるデータ処理の分野では依然として専用回路によってデータ処理装置を実現することが主流となっている。しかしながら、一般に専用回路による実現は、仕様変更や拡張などに対応するのが困難であり、仕様変更などに対しては、再設計やアーキテクチャ変更を行っているというのが実状である。したがって、専用回路を用いたデータ処理装置は、開発期間を含む開発コストや製品価格が高くなるという大きな問題を抱えている。

#### 【0006】

【発明が解決しようとする課題】演算機能などを内蔵したプロセッサーを用いたソルーションに対し、ダウンロードブルのマイクロコード（マイクロプログラム）を用いることにより処理速度を向上することができる。また、このシステムでは、プログラムを変更することによって仕様変更や拡張などに対応することができる。

【0007】しかしながら、マイクロコードによりリアルタイムな処理を実現しようとすると、復号化処理などで通常必要とされる複数のリアルタイム処理を実行するときに、マイクロプログラムを切り替える必要が生ずる。このときの切替時間を短縮するためには、複数のマイクロコードを格納可能なメモリを別に用意しこれをセレクタなどで必要に応じて選択する方法を採用することが可能である。この方法は、高速処理の観点からは効果的な方法であるが、リアルタイム処理の種類とその数が増加するに連れてあまり魅力的な方法では無くなる。即ち、処理の種類および数量に対応した多数のメモリを用意する必要があり、また、それらを切り替えるために膨大なセレクタが必要となる。したがって、そのような膨大なアーキテクチャを現実に動作させようとすると動作周波数あるいは回路規模いずれの観点からも問題となる。さらに、共通性のある処理が存在しても多重化できないので、結果的に無駄なマイクロコードを多く抱えるシステムとなり経済的でもない。

【0008】経済的なシステムとするためには、マイクロコードを用いたプログラムで必要な処理要因を判定し、その処理に必要なマイクロコードをロードし直すという方法がある。しかしながら、この方法は、判定、待避およびロードなどのプログラムを開始するための処理に時間を費やし応答性を低下させる。このため、本来の目的であるリアルタイム応答を犠牲にすることになるので、許容できない。

【0009】専用回路で実行される特殊な処理を適当な単位でマイクロプログラム化して実行するとと共に、その特殊な処理を実行中に、特殊な処理以外、たとえば、エラー処理あるいはコミュニケーションなどの汎用的な

処理が必要とされるときに割り込み要求を発させ、割り込みに応答する形でマイクロプログラムを切り換えるという方法もある。しかしながら、この方法でも応答性が問題となる。すなわち、専用回路に相当する特殊な処理を実行中に、外部からほかの要因で割り込みが発生すると、現在実行中の命令を中断するか、命令実行が完了するまで待つ必要がある。命令を中断するには、その時のレジスタ状態を保存する必要があるので、リアルタイムでの応答が必要な処理を行っているにも関わらず、レジスタへの待避あるいはロードに最低でも数クロックのオーバーヘッドが発生する。また、命令実行が完了まで待つことは、必要とされる汎用命令を処理するタイミングが遅れることを意味する。

【0010】これの解決策としては、マイクロプログラムで動作する処理系をマイクロプログラム単位で複数用意する方法がある。しかしながら、似通った構造を持つ処理系を複数用意する必要があるために、回路規模が大きくなる。さらに、複数の処理系で行われている処理の同期を取るために余分な回路や命令も必要となり、演算ユニットを内蔵したプロセッサと同等の処理系が複数必要になったり、それを動かす複数のマイクロプログラムを並列処理に適したものにする必要がある。したがって、この方法も回路規模、プログラム開発などの点で経済的ではない。

【0011】マイクロプログラムの処理系を複数設けるかわりに、マイクロプログラムの処理系とは別にこの処理系から制御される専用回路を設け、専用回路もマイクロコード命令で制御する方法もある。しかしながら、専用回路がマイクロプログラムによる汎用処理の命令コードの組み合わせによるプログラムのために、割り込みなどに対しても同時並列的な処理ができず、レジスタのロード・ストアによる制御となり結果的に大きなオーバーヘッドとなる。したがって、リアルタイムな高速処理を実現することが難しい。

【0012】このように、マイクロコードを用いた制御装置は、比較的高速でフレキシビリティがあるのでリアルタイムな処理を行うデータ処理装置としては魅力的なものである。にもかかわらず、リアルタイム処理に略完全に対処できるアーキテクチャはまだ提案されていない。

【0013】さらに、本発明の前提として、マイクロコードについてさらに幾つかの問題を指摘しておく必要がある。従来のマイクロコードを用いた制御では、制御が単純になるという理由から固定長命令が良く使用されてきた。これは処理速度を向上する上で大事なことではあるが、コード効率の点ではあまり良い選択肢とは言えない。実際、演算機能も備えたマイクロプロセッサー等のより大きなデータ処理装置では、可変長命令もコード効率の観点から伝統的に使用してきた。一方、可変長命令はコード効率の点では良い選択肢といえるが、プロセ

ッサーのスループットを上げるために複雑なブリュエット制御を行ったり、命令コードキューを追加する必要があるなどの理由から、高速処理を要求されるマイクロプロセッサーでは敬遠されている。

【0014】さらに、マイクロコードの効率を上げるためにには、分岐命令や条件分岐、サブルーチン・コール、リターン、割り込み処理要求サービスへの分岐と復帰、スタック処理などが命令レベルでサポートされていることが望ましいことは良く知られている。しかしながら、これらの処理は数クロックを費やすので、1クロック毎の制御を問題とするリアルタイム処理には適用が難しく、単にオーバーヘッドを増加させる要因とされてきた。すなわち、マイクロコード制御により高速応答性を追求することと、マイクロコードのプログラミングの生産性を上げる分岐命令やコール、リターン命令（サブルーチンからの復帰以外の割り込み処理からの復帰命令なども含む）を採用することとの両立は困難であった。

【0015】現在の高性能マイクロプロセッサで採用されているスーパーパイプライン技術やスーパースケーラ技術は、動作周波数を上げたり、データ処理のスループットを上げることにより、復帰処理などに係るオーバーヘッドの問題を平均命令実行数という観点ではある程度緩和している。しかしながら、これらのテクノロジーは分岐命令、サブルーチンコールあるいは割り込み処理などにおけるペナルティを本質的に解決している訳ではない。すなわち、現在の高性能マイクロプロセッサーでも、1クロックを問題とするようなリアルタイム制御に適用するには問題が多く本質的に向いていない。これは、その回路規模が膨大であり、コスト的にも見合わないというだけでなく、その制御構造に大きく依存している本質的な問題である。

【0016】そこで、本発明は、従来技術では専用装置でしかその実現が難しいとされてきた分野へも適応可能な高速でフレキシブルな制御装置およびデータ処理装置を実現することを目的としている。特に、本発明は、アプリケーション毎に対応する専用の1つまたは複数のマイクロプログラムと、汎用データ処理用のマイクロプログラムとを簡易な方法により多重処理あるいは並列処理を可能とするマイクロアーキテクチャ、それに適したマイクロプログラム、さらに、制御方法を提供することを目的としている。

【0017】さらに、本発明は、そのマイクロアーキテクチャをベースに、1クロック処理単位でのリアルタイム応答および制御の要求される分野へ、従来技術では適用が困難とされてきた多重処理や割り込み処理を可能とした制御装置およびデータ処理装置を提供することを目的としている。

【0018】また、本発明においては、マイクロプログラムによる高速制御が可能な制御装置およびデータ処理装置であって、アプリケーション毎に専用処理が実現可

能であり、さらに、変更や拡張などに対応可能であると共に、コンパクトおよび低成本で実現可能な制御装置、マイクロプログラムおよび制御装置、さらにその制御装置を備えたデータ処理装置を提供することを目的としている。

#### 【0019】

【課題を解決するための手段】このため、本発明の制御装置は、特定のデータ処理に適した専用回路に対する専用命令をデコードおよび実行可能な少なくとも1つの専用データ処理ユニットと、専用命令以外の処理を行う汎用命令をデコードおよび実行可能な汎用データ処理ユニットと、マイクロコード化された専用命令および汎用命令を記憶したコードメモリ（コードROM、コードRAM）から命令コードをフェッチし、専用データ処理ユニットおよび汎用データ処理ユニットに供給するフェッチユニットとを有している。また、本発明のデータ処理装置は、このような制御装置（コントロールユニット）と、この制御装置により制御可能な少なくとも1つの専用回路とを有している。

【0020】まず、本発明の制御装置においては、特定のデータ処理に適した専用回路のための1つまたは複数のマイクロコード制御系と、汎用的なマイクロコード制御系とを別々に備えている。さらに、本発明の制御装置は、これらの制御系に対し共通のフェッチユニットを備えている。したがって、本発明の制御装置を備えたデータ処理装置においては、汎用処理、およびこの汎用処理と処理内容の異なる1つまたは複数の専用回路を用いた処理が多重処理可能となり、その処理のための同期制御を命令コードレベルで実行できる。

【0021】すなわち、本発明の制御装置は、専用データ処理ユニットを備えているので、符号化されたデータストリームからスタートコードを検出したり、可変長符号化されたデータを連続デコードするなどのリアルタイム性が要求され、専用のハードウェアにより実行した方が高速化し易い処理を、専用命令に基づき、その処理に適した専用回路により処理させることができる。

【0022】それと共に、汎用データ処理ユニットを備えているので、汎用命令による処理とは完全に分離して専用命令を実行できる。したがって、データ処理装置内部、あるいは外部から応答要求があった場合などに対応する汎用命令による処理を専用回路を停止せざることなく実行できる。このため、専用回路のリアルタイム性を犠牲にすることなく汎用命令を実行できる。

【0023】さらに、専用データ処理ユニットおよび汎用データ処理ユニットに共用あるいは共通のフェッチユニットが用意されている。このため、専用データ処理ユニットおよび汎用データ処理ユニットにおいては、共通のマイクロプログラムから共通のプログラムカウンタに基づきフェッチされる命令コードに従って順次処理が行われる。したがって、命令コードレベルで専用データ処

理ユニットおよび汎用データ処理ユニットの処理を正確に制御できる。このため、専用回路と汎用データ処理系の間で同期を取るための通信回路あるいはその処理のために割かれるレジスタのロード・ストア処理などの時間は不要となる。その結果、多重処理のためのオーバヘッドが発生せず、回路規模を大きくする必要もなくなる。さらに、専用回路と、汎用データ処理系とを同期制御するプログラムをそれぞれ開発する必要もないので経済的である。

【0024】そして、本発明の制御装置、およびその制御装置を備えたデータ処理装置においては、専用データ処理ユニットおよび汎用データ処理ユニットを共通して制御するマイクロプログラムを変更することにより仕様変更あるいは拡張などにフレキシブルに対応することができる。

【0025】さらに、専用回路部分の処理を専用命令にて実行し、その専用命令を汎用命令と同じレベルでフェッチするようにしている。このため、マイクロコード処理系を介して専用回路を動作させるようなアーキテクチャでは回避できない、レジスタに専用回路用の命令コードをロード・ストアするためのオーバーへッドは存在しない。したがって、専用回路と汎用データ処理系との間で同期制御のためのタイムラグは事実上発生しない。

【0026】このように、本発明の制御装置では、専用回路を制御するマイクロコード（専用命令）を備えた1つまたは複数の専用マイクロプログラムと、汎用データ処理を行うマイクロコード（汎用命令）を備えた汎用データ処理マイクロプログラムとを1つのフェッчユニットにより扱うことが可能である。このため、本発明の制御装置は、幾つかの共通モジュールおよび／あるいは専用モジュールによる多重処理を簡単に実現できるアーキテクチャを備えている。したがって、この制御装置を備えた本発明のデータ処理装置は、1クロック単位でのリアルタイム応答性が要求される専用回路の処理を継続しながら、多重処理、割り込み処理、さらには、他の専用回路の専用命令を実行することができる。また、専用命令を実行させながら汎用命令を並列実行可能である。そして、データ処理装置全体をコンパクトに纏め、低コストで提供することもできるという特徴を有する。

【0027】フェッчユニットでフェッчされる命令コードは専用命令または汎用命令であり、そのいずれであるかを、専用データ処理ユニットおよび汎用データ処理ユニットのデコーダにより、命令コードに含まれるオペレーションコードをデコードすることによって判別することも可能である。しかしながら、そのような判別はデコード遅延を招く。それと同時に命令体系（命令コード体系）も一元化されてしまい拡張性の点からも好ましくない。そこで、命令コードに専用命令および汎用命令、さらに複数の専用データ処理ユニットを有するときはいずれかの専用データ処理ユニットの専用命令であるかを

識別可能な識別コードを設けることが望ましい。この識別コードにより、デコード遅延を最小とし、専用命令の追加や拡張が容易にできる。さらに、フェッчユニットに、識別コードをデコードして汎用データ処理ユニットまたは1つあるいは複数ある場合はいずれかの専用データ処理ユニットに対しデコードおよび実行を指示する実行制御部を設けておくことが望ましい。

【0028】識別コードをデコードし、汎用データ処理ユニットまたは専用データ処理ユニット、さらに複数の専用データ処理ユニットを有するときはそれらのいずれかに対しフェッчした命令コードをデコードおよび実行するように指示する工程を設けることにより、専用データ処理ユニットおよび汎用データ処理ユニットにおいては不要な命令コードをレジスタにロードあるいはストアする処理を防止できる。さらに、専用データ処理ユニットおよび汎用データ処理ユニットにおいて、不要な命令コードをデコードすることによる回路上のクリティカルパスの発生を防止できる。

【0029】さらに、専用データ処理ユニットまたは汎用データ処理ユニットにおいて命令コードをデコードし、専用データ処理ユニットおよび汎用データ処理ユニットにおける処理を並列実行可能であるか否かを判断する方式を採用することも可能である。しかしながら、この判断処理はハードウェアの増加を招き、また、それは新たなクリティカルパスを招き、事実上動作周波数にも影響を与える可能性がある。

【0030】そこで、命令コード中に、後続の命令コードが汎用データ処理ユニットまたは専用データ処理ユニット、さらに複数の専用データ処理ユニットを有するときはそれらのいずれかで実行されるべきか、あるいは、汎用データ処理ユニットにおいて並列処理可能であるかどうかを明示する並列処理フラグを設けることで、これらの問題を回避することができる。並列処理フラグを設け、命令コードの並列処理フラグに基づき専用データ処理ユニットのいずれか、または汎用データ処理ユニットに対し後続の命令コードのデコードおよび実行することを指示する工程を設けることにより、命令コードをデコードして並列処理（多重処理）が実行可能であるか否かを判断する処理を省くことができるので、オーバーへッドが発生するのを防止できる。

【0031】さらに、制御装置は、汎用データ処理ユニットまたは専用データ処理ユニット、さらに複数の前記専用データ処理ユニットを有するときはそれらのいずれかのための実行レジスタを有し、フェッчユニットは、実行レジスタに命令コードをストアすると同時に後続の命令コードをフェッчできるようにするフェッчデータ（命令コード）出力部を備えていることが望ましい。特に、少なくとも専用データ処理ユニットのための実行レジスタを備えていることが望ましい。そして、複数の専用データ処理ユニットを有する場合は、各々の専用データ

タ処理ユニットのための実行レジスタを有することがさらに望ましい。また、命令コードの並列処理フラグに基づき、専用データ処理ユニットのいずれか、または汎用データ処理ユニットに対し後続の命令コードのデコードおよび実行を指示する実行制御部を設けておくことは極めて有効である。

【0032】実行レジスタに命令コードをストアすると同時に後続の命令コードをフェッチする工程を設けることにより、専用データ処理ユニットおよび汎用データ処理ユニットに常に1クロックで命令コードを供給することができる。同時とは、同一のクロックサイクルまたは次のクロックサイクルの立ち上がりまたは立下りのタイミングまで含めた範囲であり、1クロックで処理が進む程度のタイミングを示している。したがって、本発明のデータ処理装置においては、実行するのに数クロックが必要となるスタートコードサーチ命令による処理を行っているときでも、1クロックで実行し、1クロックでフェッチするという処理を基本的に続けることができる。このため、処理を1クロック単位で正確に定義でき、実行することが可能である。したがって、多重処理あるいは割り込み処理などを専用命令に対して並列に制御可能となる。

【0033】マイクロプログラムの命令コードは、コード長（ワード長）が可変であればコード効率は良い。しかしながら、プリフェッチバッファを用意するなどの対応を行わないと使用される命令長が変わるので、効率良く命令コードを取得できない危険性がある。また、バス幅に相当するワード境界で命令コードが2ラインにまたがって分割されていると1クロックで命令コードをロードできない。一方、プリフェッチバッファを用いると、分岐命令が発生したときなどはプリフェッチバッファに新しい命令がロードされるまで命令の実行が停止してしまう。

【0034】そこで、命令コードに、該命令コードのワード長を示す命令長コードを設け、フェッチユニットは、命令コードに続く最大命令長分のマイクロコードを後続の命令コードとしてフェッチするフェッチデータ出力部を備えていることが望ましい。さらに、フェッチユニットは、2ライン分の命令フェッチアドレスを出力するフェッチアドレス出力部を備えていることが一層望ましい。

【0035】命令長コードを設けることにより、その命令コード全体（オペレーションコード部の全ビット）をデコードしなくとも後続の命令コードの先頭アドレスが判る。このため、命令コードに続く最大命令長分のマイクロコードを後続の命令コードとしてフェッチする工程を設けておくことにより、プリフェッチバッファを設けておかなくても後続の命令コードを確実にフェッチすることができる。そして、2ライン分の命令フェッチアドレスを出力することにより、命令コードがワード境界に

またがっている場合でも1クロックで命令コードをフェッチすることができる。したがって、コード効率の良い可変長の命令コードを採用し、プリフェッチバッファを用いずに常に1クロックで所定の命令コードをフェッチすることができる。このため、この点でも1クロック単位で処理を正確に定義し実行することができるので、汎用命令で構成される多重処理プログラムあるいは割り込み処理プログラムなどを専用命令に対して並列実行制御しやすくなる。

【0036】先に説明したが、効率の良いマイクロプログラムを作成するには、サブルーチン・コール、リターンあるいは割り込み処理機能などが不可欠である。そこで、フェッチユニットに、命令フェッチアドレスを出力するフェッチアドレス出力部と、このフェッチアドレス出力部に対し1つまたは複数のテンポラルレジスタに格納された復帰アドレスを供給可能なフェッチ制御部とを設けることが望ましい。さらに、汎用データ処理ユニットには、分岐または割り込み命令を実行するときに復帰アドレスをメモリ（スタック）に記憶すると共にテンポラルレジスタにストアし、復帰処理を実行するときに次の復帰アドレスをメモリからテンポラルレジスタにロードするレジスタ管理機能を設けることが望ましい。復帰処理を実行するときに次の復帰アドレスがテンポラルレジスタにロードされていれば、次の復帰命令を処理するときにメモリからテンポラルレジスタにロードする処理が不要となる。このため、1クロックで復帰命令を処理することが可能となり、復帰関連の処理も1クロック単位で正確に定義し実行できる。したがって、この機構により多重処理あるいは割り込み処理を制御しやすいマイクロコードによる制御装置の提供が可能となる。

【0037】さらに、割り込み処理の場合は、復帰アドレスに加えてコンディションコードも1クロックでレジスタにロードできることが望ましい。このため、レジスタ管理機能は、割込み復帰アドレス用およびコンディションコード用のテンポラルレジスタを各々備えており、割込み命令を実行するときに復帰アドレスおよびコンディションコードをメモリに記憶すると共に、割込み復帰アドレス用および前記コンディションコード用のテンポラルレジスタに各々ストアし、復帰処理を実行するときに復帰アドレスのメモリアドレスおよびコンディションコードのメモリアドレスを出力し次の復帰アドレスおよびコンディションコードをメモリからテンポラルレジスタに各々ロードすることが望ましい。これにより、割り込み処理からの復帰も1クロックで行うことができる。

【0038】このように、本発明の制御装置、上述した工程を備えた制御方法、さらにはROM、RAMあるいは他の移動式（ムーバブル）、固定式、あるいはネットワークで接続された、コンピュータ、制御装置またはデータ処理装置が読み取り可能なメモリ装置あるいは記録媒体に記録されて供給されるマイクロコードにおいて

は、1クロック単位での正確な制御が可能である。さらに、分岐命令やコール命令、リターン命令もすべて1クロックで実行される。このため、正確なパイプライン制御によるタイミング設計が可能で、分岐の成立不成立にかかわらず必要なクロックサイクルは最小（1クロック）であり一定であり、ステートマシンと同様の無駄のない制御が可能となる。

【0039】これは、本発明により、多重処理可能な汎用データ処理命令を備えたマイクロプログラムによる制御であるにもかかわらず、専用のステートマシンによる制御と同一品種の処理が可能になったことを意味する。あるいは、コード効率の極めて優れた汎用的なマイクロプログラムによる制御であるにも拘わらず、それぞれ専用のマイクロプログラムを独立してもらったシステムと等価なトランスペアレントな処理が可能であることを意味する。しかも、高い性能を提供しながら、制御装置を実現するための回路として重複する部分がほとんどなく、極めてコンパクトである。

【0040】したがって、本発明のマイクロコードによる制御装置と、この制御装置を備えた本発明に係るデータ処理装置により、フレキシビリティのない専用回路だけによって実現されるデータ処理装置と同等の性能を備え、これに代わる効率的で高速なデータ処理装置を提供できる。

#### 【0041】

【発明の実施の形態】以下に図面を参照して本発明の実施の形態を説明する。図1にMPEG標準によって規格化されたビットストリームに含まれる可変長符号（VLC）を復号（デコード）する機能を備えたデータ処理装置（可変長符号復号装置、以降においてはVLD）1の概略構成をブロック図により示してある。

【0042】画像情報は、音声情報あるいは文字情報に比較し数桁程度も情報量が多く、マルチメディアを実現するためにはその多量の情報を符号化して処理することが重要となっている。MPEG標準はその1つであり、現在、MPEG-2が様々なアプリケーションに対して適用可能なものとして多く採用されている。そのビットストリーム（MPEGビットストリーム）は、階層構造になっており、上位層から順にシーケンス層、GOP（グループオブピクチャ）層、ピクチャ層、スライス層、マクロブロック層、およびブロック層を備えている。シーケンス層からスライス層まではの各層は、ユニークなスタートコードが用意されており、これを認識することによりデコードすることができる。ピクチャ層以下の各層においては、膨大な画像情報をコンパクトに符号化するために可変長符号化またはエンタロピー符号化することにより平均情報量を削減している。したがって、MPEGビットストリームは膨大な量のVLCを高速にデコードすることが重要であり、このため、高速なVLDが強く要求されている。

【0043】VLDにおける処理の要求性能は、単位時間当たりどの程度のビットストリーム処理を行わねばならないかに依存しており、これがある意味で内部アーキテクチャを左右する最大要因であり、設計上のトレードオフの最大ポイントと言える。上述したようにMPEGビットストリーム（ビデオ）のシンタックスは、スタートコードと呼ばれるレイヤ（層）を規定するユニークなコードと固定フィールド（FLC）及び可変長フィールド（VLC）で特徴付けられる。このため、以下のような処理要因がVLDにおける処理性能に大きく影響を与える。

- (1) スタート・コードの検出処理
- (2) 各制御のパラメータ
- (3) VLCの復号処理
- (4) MPEGシンタックスのエラー検出
- (5) アプリケーション・レベルでの柔軟性（ユーザ・データ対応など）
- (6) 上位レイヤへの対応などの拡張性

【0044】第1にスタート・コードの検出能力は、伝送品質が極めて安定した条件下でのビデオ再生にウエイトを置くアプリケーションではそれほど問題となることはない。しかしながら、ビデオ編集業務やゲーム等のインタラクティブな再生が想定されるアプリケーションでは非常に問題となる。この場合、ある程度大きなビット・ストリームの中から必要とするレイヤのスタート・コードを検出する必要があり、これが再生時のレスポンスに大きな影響を与えるからである。特に高品質の映像を扱う場合、つまり圧縮率をあまり高めるような符号化パラメータを選択できないような状況では、そのために取り扱うビット・ストリームの容量も大きくなるので、スタートコードの検出速度が人間（オペレータ）から見てVLDの操作性の印象を大きく左右する要因となる。

【0045】第2に各制御パラメータの設定・記憶処理は、デコードするMPEGビットストリームの量がそれほど多くない場合は比較的問題は少ないが、デコードするビットストリームの量が多く、時間的な余裕がそれほど無い場合は問題となる。また、時間的余裕がなく、同時に外部のホストCPUなどからのパラメータ・アクセス頻度が多い場合は、この処理速度がさらに問題となる。MPEG-2の場合、パラメータの種類も多くこれらをレジスタ形式で持つことはあまり経済的でないからである。この処理を専用ハードウェア（専用回路）ですべて処理しようとすると回路規模も大きくなり経済的でない。しかし、MPUで処理するには時間がかかるので、外部からのアクセスに対応しながらデコード処理を行うことは時間的に難しい。

【0046】第3にVLCの復号処理は、基本的には1つのコードのデコードが完了しないと次のVLCのデコードを開始できないという欠点を持つ。特に、デコードすべきビットストリームの量が多い場合は、デコードの

遅延が致命的な問題となり、回路規模を大きくせずにこれを解決するには回路の動作周波数を上げるほかない。しかし、動作周波数は、システム全体の設計要求をより厳しいものとし開発期間やチップのイールドに大きく影響する。回路規模を大きくしてもよければ、VLCテーブルを完全に2重に用意したり、スライス・レイヤのレベルで並列デコードするなど解決手段はいくつかある。しかしながら、このような対策は、ダイサイズに直接影響したずらにチップ単価を上昇させるだけであり好ましくない。

【0047】第4に、MPEGビットストリームは、厳密なレイヤ構成を規定しており伝送品質の問題などでこれが保障できなくなると表示画像への影響やデコードしているパラメータ矛盾等によりシステム的に破綻する危険がある。通常は、誤り訂正などの上位レイヤ・レベルでの処理で回避できるが、デコード自体もこのような矛盾したMPEGビットストリームに対してある程度耐性をもつ必要がある。もちろん、MPEG-2では、エラー・コンシールメントなどの配慮が行われているが完全ではない。特に、デコード対象となるビットストリームの多い高品質画像が要求されるアプリケーション・ユーザにとっては製品イメージに直結する大問題である。したがって、MPEGシンタックスのエラー検出機能は必須の要件であり、また出来ればFLCやVLCレベルでのエラー・チェック機能も実装されていることが望ましい。

【0048】第5に、MPEG-2は、アプリケーション・レベルでの拡張性を配慮したユーザ・データのフィールドなどが用意されている。ユーザは、アプリケーション・レベルでこれらのフィールドを利用して、文字やテキストの多重化やアニメーション・キャラクタの多重化などの要求仕様を追加することができる。したがって、将来の拡張へも対応できるようなハードウェア構成が望ましい。

【0049】最後に、半導体の集積度向上に伴い、将来上位レイヤのデコード機能も含めて集積化することが可能となる可能性が高く、これに対応できるアーキテクチャ構造が望ましい。そして、エラー訂正や暗号化、オーディオとの同期などに対しても基本ユニットや専用ユニットの拡張だけで対応できるようであれば、経済的である。

【0050】本例の可変長符号復号装置(VLD)1は、上述した様々な要求性能は、経済的な面も考慮して満足できるようにしたデータ処理装置であって、高速な可変長符号の復号処理を実現するものである。

【0051】本例のデータ処理装置であるVLD1は、図1に示すように、ビットストリームを受信する入力FIFO2と、バレルシフターとしての機能も備えたマイクロコード方式の制御ユニット(制御装置)10と、専用回路の1つであるビットストリームからスタートコー

ドを検出するスタートコード検出回路3、他の専用回路であるVLCをデコードするためのVLCテーブル4、ビットパターンをシーケンシャルに比較してビットストリーム中の任意のパターンを高速検索する専用回路である比較回路5、ホスト・インターフェース6、IQ/I DCTインターフェース7、MCPインターフェース8、外部CPUインターフェース9、および外部CPUとのソフト的なインターフェースを実現するコミュニケーション用の内部CPU19を備えており、これらがバス11により接続されている。また、制御ユニット10には、マイクロコードを記憶したROM12およびワークエリアとなるワークRAM13が接続されている。

【0052】入力ビットストリームは、外部からの直接入力しても良いし、一度外部メモリーでバッファリングしたもの再度入力しても良い。外部メモリー・インターフェースの設計がタイトでなく、ビットストリームの供給に対しては高いレスポンスが保証できれば、内部FIFOの大きさは極めて少なくて良い。この場合、回路を小さくして十分間に合うことになる。逆に、外部メモリーのバンドはバンド幅が十分でなかったり、バス・アービトレーションによる切換え時間を大きくして、即ちレイテンシーをある程度大きくして、必要最低限のバンド幅を確保するタイトなアーキテクチャの場合は、内部FIFO2の大きさを増やして、これに対応することが可能である。内部FIFOの大きさは、入力データのレスポンスやレイテンシーの大きさに依存する。大容量のFIFO等ライブラリ・レベルで対応できない場合であっても、シングル・ポートRAMを利用して大容量FIFOを実現可能なので、全体処理の破綻が発生しない範囲であれば、入力ビットストリームの条件はかなり悪くても良く、ほとんどの場合全く問題ない。

【0053】制御ユニット10は、内部CPUで直接制御可能なインテリジェント型のバレルシフターを備えたものであり、制御ユニットの構成については以下で更に詳しく説明する。この制御ユニット10は、スタートコード検出や特定のビットパターン検出、VLCとFLCのデコードといった専用回路を用いた処理と、指定領域へのデータ格納処理、エラー処理、割り込み処理などの汎用処理(汎用データ処理)をマイクロコードで記述されたプログラムにより制御できるようになっている。

【0054】スタートコード検出回路3は、64ビット単位で動作するもので検出に最適なビット・シフト量をデコードして、これをバレルシフターに通知するものである。MPEGビデオの場合、スタート・コードはユニークであり、これを検出するための最適シフト量を決定するだけで良い。バレルシフターは32ビット形式であり、バレルシフター内部のレジスター側の情報をデコードし最大64ビットの高速シフトを実現している。さらに、バイト・アラインは前提としなくても良いので原理的にスタート・コード検出漏れの危険は全く無い構成と

なっている。

【0055】VLCデコーディング・テーブル4は、MPEG-1とMPEG-2とで使用されるVLCデコーディング・テーブルから構成されており、テーブルの選択情報はパレルシフター機能を備えた制御ユニット10から得る構成となっている。

【0056】ホスト・インターフェース6は、内部CPU19の内部メモリへのアクセスを許すもので内部CPU19の実行プログラムの構成によっては、データ処理や復号処理の制御も可能となる。例えば、ゲームなどのようなアプリケーションで高速にシーンチェンジを繰り返して、刻々とストーリ展開を図ることも容易に実現可能である。これらは、高速なスタートコード検出機能と細かいクロック単位での復号処理の制御が必要であるが、本例のVLDではリアルタイム性の要求されるこのような処理も実行可能であり、外部CPUとのコミュニケーションにより実現することができる。

【0057】IQ／IDCTインターフェース7は、逆量子化器或いは逆離散コサイン変換器とのインターフェースである。内部CPU19がFIPU（2命令／CLK）のように高速なCPUであれば、逆量子化処理まで実行して、直接逆離散コサイン変換器へデータを転送することが可能である。

【0058】MCPインターフェース8は、動き補償処理に必要なマクロブロック・アドレスやタイプ、動きベクトル情報などをMCP（動き補償器）へデータ転送するものである。マクロブロック・アドレスは、MCP（動き補償器）の都合により、エスケープ・コードも含めてインクリメント・アドレスではなく、絶対アドレスに変換して転送することも可能である。

【0059】内部CPU19は、VLD1としては特に指定すべき機能はなく、ユーザの集積化の都合に依存する。CPU19の処理性能は低くても良く、実際にVLDで実現する用途に応じて選択の幅が広い。CPU性能の低いものを選択した場合、逆量子化処理等をソフトウェアで対応できないこともある。この場合は、専用ハードウェア・モジュールの追加が必要となる。また、将来的拡張性を重視する場合は、仕様変更あるいは追加に対して柔軟に対応できるように、ある程度の性能のCPUを内蔵しておくことが望ましいことはもちろんである。

【0060】図2に、VLD1の制御ユニット10の概略構成をブロック図により示してある。本例の制御ユニット10は、コードROM12に記憶されたマイクロコード（マイクロプログラム）によって動作する制御装置であって、命令コードをフェッチするフェッチユニット(FU)21と、命令デコードと実行を行う命令デコード実行ユニット(DXU)22と、バス11を介して外部の専用回路あるいは他のリソースとインターフェースするためのインターフェースユニット(IFU)23の3つ

10

の大きなブロックを備えている。

【0061】命令デコード実行ユニット22およびインターフェースユニット23は、機能的に、スタートコード検出回路3などの専用回路で処理する専用命令をデコードおよび実行する専用データ処理ユニット（以降においてはVU）31と、それ以外の汎用のデータ処理命令（汎用命令）を実行する汎用データ処理ユニット（以降においてはPU）32とに別れている。したがって、VU31で可変長符号復号処理（VLD）にかかる専用命令を実行しながら、PU32により別の汎用データ処理を並列実行することが可能な構成となっている。また、本例の制御ユニット10は、1クロックで1命令の高速実行を可能とするために、デコードサイクルと実行サイクルを同じフェーズで行う形式を採用している。結果として、これらのモジュールDXU22およびIFU23はVU31およびPU32にシェアされた構造となっている。

20

20

30

40

40

【0062】独立した機能を有するVU31およびPU32が構成されるために、命令デコード実行ユニット22は、サブモジュールとして専用命令をデコードするユニット(VDEX)33と、汎用命令をデコードするユニット(PDEX)34とを備えている。さらに、デコードされた専用命令を実行するユニット(VDR)35と、デコードされた汎用命令を実行するユニット(PDR)36を備えている。また、インターフェースユニット23は、サブモジュールとしてバス11にアクセス可能なサブユニット37および38を備えている。このように、VU31は、独自のデコードユニット33、実行ユニット35およびインターフェースユニット37を備え、また、PU32も独自のデコードユニット34、実行ユニット36およびインターフェースユニット38を備えている。したがって、VU31およびPU32は、独立して命令をデコードし実行可能である。しかしながら、本例では、並列処理の指示がプログラムに記述されていない限り並列動作には移行しないようになっている。

【0063】VU31とPU32に共通して参照できる1つまたは複数のレジスタCRが用意されておりレジスタレベルで相互にデータ転送が可能となっている。そのため、専用命令と汎用命令を組み合わせることによりいろいろなデータ処理に柔軟に対応できる。さらに、図2では、VU31は1系統であるが、異なった専用回路を用いて多重処理ができる処理系であれば、VUを複数系統設けることも可能であり、本明細書で説明する構成および処理を同様に適用することができる。

【0064】一方、これらのVU31およびPU32に対する命令コードをフェッチするフェッチユニット21は、コードROM12に対し命令フェッチアドレスを出力するフェッチアドレス出力ユニット(VFAU)41と、命令コードをコードROM12からフェッチしフェッチレジスタFR0、FR1およびFR2に出力するフ

エッチデータ出力ユニット（VFRU）42と、フェッチ制御や命令のデコード実行制御を行う実行制御ユニット（VPFD）43と、割り込み制御や例外処理制御、リセットからのフェッチ、リターン命令実行後のフェッチ制御を行うフェッチ制御部（VPIRQ）44の4つのユニットを備えている。フェッチされる命令コードは、動作周波数の上限を上げるために単純化された命令セットが用いられており、さらに、フェッチユニット21の内部制御が命令コードの最初の数ビットをデコードするだけで行えるようなフィールド構成となっている。

【0065】図3に、本例の制御ユニット10で採用されているマイクロコードの構成を示してある。本例では、1ワード16ビットとして最大長が3ワードのマイクロコードが命令コード50として採用されている、命令コード50の先頭は、この命令コード50と共に後続の命令コードを並列処理可能であるか否かを示す1または数ビットの並列処理フラグ51のフィールドである。これに続いて、命令コード50のワード長を示す1または数ビットの命令長コード52のフィールドが設けられており、さらに、命令コードのディステイネイションを示す1または数ビットの識別コード53のフィールドが設けられている。これらに続いて、オペランドおよびパラメータなどの処理内容を指示するフィールド54が設けられている。本例の制御ユニット10では、可変長命令コードが採用されておりオペランドおよびパラメータの長さによって2ワードまたは3ワードの命令コードが混在している。

【0066】まず、識別コード53は、命令コード50がPU32でデコードおよび処理される汎用命令か、VU31でデコードおよび処理される専用命令であるかを示すデータである。本例では、汎用命令は識別コード53が“0”にセットされ、専用命令の識別コード53が“1”にセットされている。さらに、複数のVU系が用意されている場合は、特定のVU系を示す識別コードをセットすることによりデコードおよび処理されるVU系を区別することができる。

【0067】この識別コード53は、フェッチユニット21の実行制御ユニット43でデコードされ、VU31またはPU32のデコードユニット33または34にデコード指示が発せられる。これにより、デコードユニット33または34はフェッチレジスタFR0ないしFR3にロードされた命令コードをデコードする。したがって、VU31またはPU32は、専用命令または汎用命令だけをそれぞれデコードし実行する。このため、他の命令をデコードする処理に時間が費やされることはなく、それによるデコード遅延を防止できる。また、VUまたはPUにおいては専用命令または汎用命令だけをデコードおよび実行するので、各々の処理ユニットにおける処理内容を1クロック単位で制御するように記述で

き、またそのように制御することが可能となる。

【0068】命令長コード52は、個々の命令コード50のワード長を示すものであり、本例においては命令コードが2ワードまたは3ワードであることを示すために用いられる。フェッチユニット21においては、命令長コード52をフェッチアドレス出力ユニット41でデコードすることにより、後続の命令コードの先頭アドレスが命令コード全体をデコードしなくても判明する。したがって、命令コード50がVU31またはPU32でデコードされなくても後続の命令コードを常にフェッチすることができる。このため、フェッチされた命令コードを次のクロックのタイミングでフェッチレジスタFR0ないしFR2にストアすることによりフェッチサイクルを1クロック単位で行うことが可能となる。さらに、可変長の命令コードをフェッチするためのプリフェッチ制御およびプリフェッチバッファは不要となる。

【0069】並列処理フラグ51は、後続の命令コードが多重処理できることを示すデータである。このコード51により、多重処理が可能であることが指示されている場合は、フェッチユニット21のフェッチデータ出力ユニット42は、命令コードがVU31またはPU32でデコードされるのを待たずに、フェッチしている次の命令コードをレジスタFR0ないしFR2にストアする。そして、実行制御ユニット43により、識別コード53にしたがってPU32またはVU31にデコードおよび実行させることができる。したがって、多重処理の可否を命令コードレベルで明確に指示することができる。また、多重処理の可否をVU31またはPU32のデコードユニット33または34などで相互の処理状況に基づき判断しなくても良い。したがって、多重処理を行うためのオーバヘッドおよびその判断を行うためのハードウェアによる回路規模の拡大、回路上のクリティカルパスの発生を防ぐことができる。

【0070】このように命令コード50の先頭に並列処理フラグ51、命令長コード52および識別コード53を設けておくことにより、フェッチユニット21においては、命令コードの先頭、たとえば、フェッチレジスタFR0の内容をデコードするだけで内部制御を行うことができる。

【0071】本例においては、専用命令に続く汎用命令が多重処理可能であるか否かが命令コード中に記述される。たとえば、フェッチレジスタFR0ないしFR2にストアされた命令コードが専用命令であり多重処理が可能であると、その命令コードはVU31のデコードユニット33でデコードされると共に実行レジスタVR0およびVR1にストアされる。本例においては、専用命令としては2ワードの命令コードが用意されており、2ワード分の実行レジスタVR0およびVR1で専用命令をストアできる。また、専用命令がフェッチレジスタFR

0ないしFR2にストアされると共に、命令長コード52がフェッチアドレス出力ユニット41でデコードされているので、プログラムカウンタ（インストラクションカウンタ）IPは後続の命令コードの先頭まで進み、後続の命令コードがフェッチされている。そして、専用命令が実行レジスタVR0およびVR1にストアされるとき、実際には次のクロックサイクルの立ち上がりまたは立ち下がりのタイミングに、後続の命令コードがフェッチレジスタFR0ないしFR2にストアされる。そして、その命令コード（汎用命令）がPU32のデコードユニット34でデコードされ、汎用命令に基づく処理が専用命令に基づく処理と並列に実行処理される。

【0072】この際、プログラムカウンタIPは、さらに後続の命令コードの先頭に進み、その命令コードがフェッチされているので、フェッチレジスタFR0ないしFR2にストアされた命令コードが消費されると同時に後続の命令コードがフェッチレジスタにストアされる。このように、本例の制御ユニット10においては、フェッチサイクルが1クロック単位で繰り返される。さらに、専用データ処理ユニットVU31のために実行レジスタVR0およびVR1が用意されているので、専用命令を専用回路において実行される処理が終了するまで、そのVU31のレジデンントな命令、継続して制御する命令あるいは繰り返して実行する命令とすることができる。

【0073】すなわち、専用回路で行うデータ処理は一連のデータ群、たとえばビットストリームに対し同じ処理を繰り返して行うことが多い。したがって、次の命令ができるまで、命令がクリアされるまで、あるいは終端条件が整うまでは、いったん実行レジスタにストアされた命令に基づき処理を進めることができが望ましいケースが多い。本例の制御ユニット10においては、実行レジスタを設けることによりそのような専用回路における処理を行なながら、それとは異なるあるいは関連した、他の専用回路あるいは汎用処理系における当座あるいは緊急性を必要とする処理を並列して行うことができ、その制御も容易となる。複数の専用回路に対し複数のVUが要求される場合は、それぞれのVUに対応した複数の実行レジスタVRを用意しておくことが望ましい。

【0074】さらに、VU31およびPU32に対し共通のフェッチユニット21が用意されており、この1つのフェッチユニット21でフェッチされた命令コードによりVU31およびPU32が制御される。したがって、VU31およびPU32の処理を多重処理も含めて命令コードレベルで1クロック単位で厳密に制御することができる。このため、VU31およびPU32の同期制御はプログラムで行うことができ、VU31およびPU32により多重処理を行うために通信回路や調停のための処理時間は不要である。そして、VU31およびPU32が共通のマイクロプログラムで制御されるので、

プログラムを変更することにより、仕様変更あるいは拡張などにフレキシブルに対応することができる。

【0075】また、VU31を制御する専用命令をマイクロコードそのもので用意してあるので、マイクロコードを介して専用回路を制御するために従来避けることができなかつたレジスタのロード・ストアによる制御のオーバーヘッドをなくすことができる。したがって、この点でもVU31とPU32を正確な同期の下でタイムラグなしに並列に処理を実行することができる。

【0076】上記にて説明したMPEGビットストリームを処理するデータ処理装置であるVLDにおける処理性能に大きく影響を与える（1）から（6）の要因は、専用回路を用いて高速性を追求することと、マイクロコードあるいはさらに大きなプログラムで作動するプロセッサを用いて汎用性を追求することのトレードオフとなる要因である。これに対し、本例の制御ユニット（制御装置）10およびこれを用いたVLD（データ処理装置）1では、専用回路と汎用処理系（汎用処理システム）とを極めて簡単な構成で多重化処理できるようにしている。このため、上記（1）から（6）の要因を設計上のトレードオフとしてではなく、専用回路を用いた高速性と、プロセッサの汎用性を兼ね備えた極めてスマートな形でクリアすることができる。そして、回路規模は大きくならず、さらに、回路設計あるいはプログラム開発のために費やす時間および費用も削減できる。したがって、コンパクトで経済的にもメリットの大きな高速VLDを提供することが可能となる。

【0077】このようなデータ処理装置1において、さらに厳密に制御を行いリアルタイム性の高いマイクロコードによる制御装置を実現するためには、1クロックでほとんどすべての制御が行えることが重要である。このため、本例の制御ユニット10は、さらに、次のような機能を備えている。

【0078】本例の制御ユニット10においては、マイクロコードの効率を上げるために可変長命令形式を採用している。しかしながら、図4に示すように、可変長命令形式は、命令実行により消費される命令長が可変であるため、コードROM12と制御ユニット10を接続するバス幅（16ビット×4、ワード境界）にまたがってアドレッシングされている（たとえば命令コードIC（n+1））と、1クロックでフェッチしフェッチレジスタFR0ないしFR2にストアすることができない。このような問題を解決するために従来用いられている手法はプリフェッチを行いコードキュー（プリフェッチバッファまたはプリフェッチレジスタ）と呼ばれるレジスタに命令の実行に先立って命令をロードしておくことである。しかし、これは、プリフェッチポインターが命令分岐発生時にプリフェッチバッファがフラッシュすると分岐後に実行すべき命令コードがロードされるまで処理が実質的に停止してしまうという問題を抱えている。こ

れを回避するには、プリデコーダやディレイデド・ブランチなどの方法が考えられているが、通常のプリフェッチ動作とは別の制御が必要となり、少なくともプリフェッチレジスタなどは必要不可欠であり、回路規模はさらに大きくなる。

【0079】このため、本例の制御ユニット10においては、フェッチャユニット21のフェッチャアドレス出力ユニット41から常に2ライン分のフェッチャアドレスFAおよびNAを出力するようにしている。これにより、ワード境界を跨いで記憶されている命令コードであっても必ず1クロックでフェッチャすることが可能である。さらに、本例では、個々の命令コードが命令長コード52を備えているので、後続の命令コードの先頭アドレスを判断することができる。さらに、最大命令長は3ワードになっており、バス幅、すなわちワード境界よりも短い。したがって、命令コードがワード境界にあるか否かに関わらず、常に、先頭アドレスから3ワード分のコードをフェッチャし、フェッチャレジスタにストアすることができる。

【0080】図5に、そのような制御を行う回路例を示してある。命令コードはワード毎に4つのコードROM 12a、12b、12cおよび12dにメモリインタリープされており、2ラインのアドレスFAおよびNAは、各コードROM毎に命令長コード52をデコードした結果に基づくフェッチャアドレス出力ユニット41からの信号φaによって選択される。さらに、各コードROM 12a、12b、12cおよび12dから各々のバスにフェッチャされたデータ出力A、B、CおよびDは、選択信号φaに基づいて選択され、フェッチャレジスタFR0、FR1およびFR2にストアされる。したがって、フェッチャレジスタFR0ないしFR2には、プログラムカウンタIPが示す先頭アドレスから3ワード分のデータが常にストアされる。このため、命令コードが可変長であるにも拘わらず、プリフェッチレジスタを必要とせず、ワード境界を跨いで配置されている命令コードであっても通常のフェッチャ動作により、たとえ分岐時であっても1クロックでフェッチャすることが可能である。したがって、フェッチャ動作のペナルティなしで、高速実行を保証できる。

【0081】このようなフェッチャ処理は、命令コードの最大長が3ワードに限られるものではなく、最大長が4ワード以上の場合にでも適用できる。ただし、5ワード以上の場合は、命令コードが整列されるコードROM、あるいはこれをダウンロードするコードRAMにおいて最大長と同じまたはそれ以上がワード境界となるように並べておく必要がある。

【0082】次に本例の制御ユニット10において、リアルタイム応答性を高めるために採用されている方法について説明する。本例の制御ユニットにおいては、全ての命令を1クロックで完了させることが重要であること

は説明した通りである。このため、コール命令やリターン命令、割り込み処理発生時の分岐とそれからの復帰に伴うスタック操作などを1クロックで実行可能なようにしている。従来、コール命令、リターン命令あるいは割り込み処理の発生後に復帰するときは、スタックに格納された復帰アドレスが用いられる。その復帰処理においては、復帰アドレスがスタック（メモリ）に格納されているので、復帰アドレスを入手するために1クロック必要とし、さらに、復帰アドレスを出力するのに1クロックを必要とする。したがって、少なくとも復帰処理に2クロック必要となってしまう。

【0083】このため、本例の制御ユニット10においては、PU32の実行ユニット36に復帰アドレスを管理する専用のポインターとテンポラルレジスタTRを別に設け、これらをレジスタ管理機能39で管理することにより、この問題を回避している。即ち、復帰アドレスをスタックに格納する際このテンポラルレジスタTRにも最新の情報として格納しておく。これにより、テンポラルレジスタTRに格納されたアドレスへ復帰する処理では、フェッチャ制御ユニット44がテンポラルレジスタTRのアドレスをフェッチャアドレス出力ユニット41に供給して実行することができ、スタックからアドレスをロードする必要がない。したがって、復帰処理を1クロックで行うことができる。さらに、その復帰処理によってテンポラルレジスタTRのアドレスを消費したときに、次の復帰アドレス情報をこのテンポラルレジスタにロードしておく。これにより、次の復帰処理も1クロックで行うことができる。

【0084】さらに、割り込み処理においては、復帰アドレスIPに加え、フラグ情報（コンディションコード）CCが必要となる。したがって、本例のレジスタ管理機能39では、これらに専用のテンポラルレジスタTRを個々に用意すると共に、2系統のアドレスをスタックに対し出力することにより、同時にこれらの情報IPおよびCCを独立してリードおよびライトが可能な構造としている。

【0085】図6に、本例のレジスタ管理機能39のうち、割り込み復帰に関連した処理を模式的に示してある。ワーク用のRAM13のスタック領域13aおよび13bには、割り込み復帰アドレスIPおよびフラグ情報CCが順番にスタックされるようになっている。割り込み命令が実行されると、その割り込み命令にかかる復帰アドレスIP(n)およびフラグ情報CC(n)がRAM13のスタック領域13aおよび13bに記録されると共に、それぞれのテンポラルレジスタTR1およびTR2にもストアされる。したがって、この段階で、復帰処理が行われると、テンポラルレジスタTR1およびTR2にストアされた復帰アドレスIP(n)およびフラグ情報CC(n)がフェッチャユニット21により用いられるので、1クロックで復帰処理を実行できる。

【0086】このとき、スタック13aおよび13bをアクセスするアドレスUP-AおよびUP-Bとしては、スタック領域13aおよび13bの次（時間経過としては事前の割り込み処理における）の復帰アドレスIP(n-1)およびフラグ情報CC(n-1)のアドレスが outputされる。したがって、メモリデータバスには復帰アドレスIP(n-1)およびフラグ情報CC(n-1)が出力されている。このため、復帰アドレスIP(n)に復帰する処理を行うとときに、具体的には次のクロックの立ち上がりまたは立ち下がりでバスに出力されている次の復帰アドレスIP(n-1)およびフラグ情報CC(n-1)をテンポラルレジスタTR1およびTR2にロード（ストア）することが可能である。そして、次のクロックで復帰アドレスIP(n-1)に復帰する処理が発生しても、テンポラルレジスタTR1およびTR2のデータを用いて1クロックで復帰処理を実行することができる。割り込み復帰に限らず、他の命令、たとえば、分岐命令あるいはコール命令における復帰処理においても同様の方法により1クロックで処理することができる。なお、復帰アドレスおよびフラグ情報をダブルワードのデータとしてスタック領域に記憶することにより割り込み処理の復帰に伴いスタック領域をアクセスするアドレスを1つにすることも可能である。

【0087】本例の制御ユニット10においては、このような制御を行うレジスタ管理機能をPU32に用意しておくことにより、1クロックで分岐命令、条件分岐、サブルーチンコール、割り込み処理要求サービスの復帰処理を実行できる。したがって、1クロックを問題とするリアルタイム処理においても、効率の良いマイクロプログラムを用いることが可能であり、生産性の高いプログラミングによりリアルタイム制御を行うことができる。

【0088】このように、本例のデータ処理装置1は、1クロック単位での正確なリアルタイム制御が可能であり、プログラマビリティを保証しながら、従来の専用回路に代わりこれを並列性の高いマイクロプログラム制御によりMPEGビットストリームの復号処理を高速で実現することができる。さらに、1クロック単位での処理を正確に定義可能で、かつ割り込み処理あるいはエラー処理などの汎用処理（汎用データ処理）を専用命令と並行して実行可能である。しかも、同期の問題を命令コードレベルで完全にクリアしており、同期制御のための通信回路などは不要である。また、重複する回路などを設ける必要がないので、極めて低コストでコンパクトに実現することができる。

【0089】もちろん、上記にて説明した本発明にかかるアーキテクチャを備えた制御ユニットは、VLDに限定されず、正確なリアルタイム制御が要求され、さらに、大きなスループットが要求されるデータ処理装置、ネットワーク制御装置などの他の制御装置においても非

常に有効である。そして、多重処理可能な汎用データ処理命令を備えたマイクロプログラムによる制御であるにもかかわらず、専用のステートマシンによる制御と同一品質の処理を提供することができる。したがって、従来の専用ステートマシンが適用されている応用分野の全てに本発明にかかる制御装置の適用を検討することが可能である。

#### 【0090】

【発明の効果】以上に説明したように、本発明にかかる10マイクロコードによる制御装置は、少なくとも1つの専用データ処理ユニットと、汎用データ処理ユニットとを備えており、これらに対し共通のフェッチユニットによりマイクロコードで記述された専用命令および汎用命令を供給できるようにしたものである。したがって、命令コードレベルで専用データ処理ユニットと汎用データ処理ユニットを同期制御することが可能であり、回路規模を大きくすることなく、それらのユニットにおいて多重処理を行うことができる。このため、開発費用などを含めたコストパフォーマンスが高く、リアルタイム性が要求される処理に適した高速処理用の制御装置およびこれを用いたデータ処理装置を提供することができる。

【0091】さらに、本発明においては、1クロック単位での正確なリアルタイム制御を実現する幾つかの構成を開示しており、大きなスループットとリアルタイム性が要求される復号装置、符号化装置、通信装置、データ処理装置、専用命令実行機能およびユーザ定義命令の実行機能などを備えたマイクロプロセッサーあるいはデジタルシグナルプロセッサなど多種多様な分野に適用できる。さらに、本発明は、高速処理をソフトウェアにて実現する目的のFPGA、ゲートアレイ、エンベデッド・アレイ、スタンダードセル・ベースのLSI、フルカスタム・ベースのLSIなどにも適用されるものである。

#### 【図面の簡単な説明】

【図1】本発明にかかる制御ユニットを備えた可変長符号復号用のデータ処理装置（VLD）の概略構成を示すブロック図である。

【図2】本発明にかかる制御ユニットの概略構成を示すブロック図である。

【図3】図2に示す制御ユニットの命令コードの構成を示す図である。

【図4】命令コードがコードROMに格納されている様子を模式的に示す図である。

【図5】コードROMから命令コードをフェッチする構成の一例を示す図である。

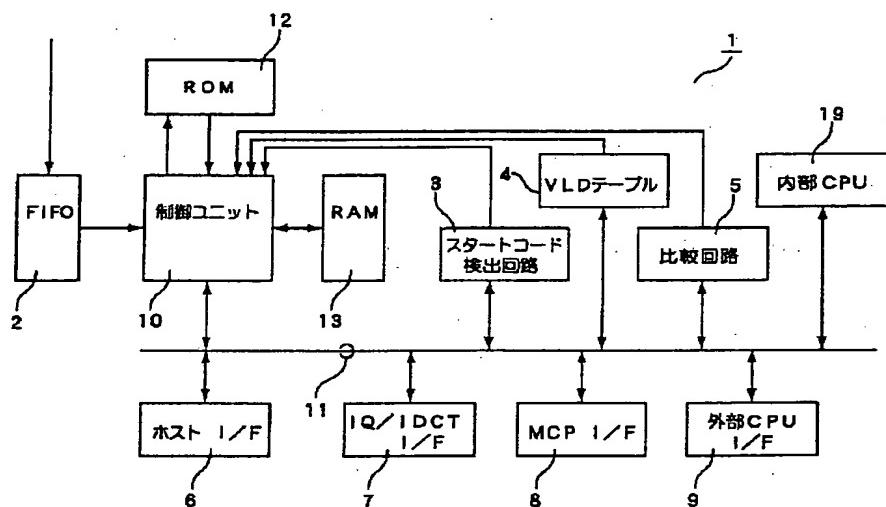
【図6】テンポラルレジスタの制御を模式的に示す図である。

#### 【符号の説明】

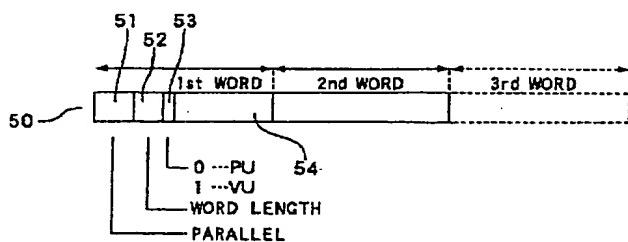
- 1 高速VLD
- 2 入力 FIFO
- 3 スタートコード検出回路

4	VLCテーブル	33、34	デコードユニット
5	比較回路	35、36	実行ユニット
6	ホスト・インターフェース	37、38	インターフェースサブユニット
7	I Q / IDCTインターフェース	39	レジスタ管理機能
8	MCPインターフェース	41	フェッчデータ出力ユニット
9	外部CPUインターフェース	42	フェッчアドレス出力ユニット
10	制御ユニット	43	実行制御ユニット
11	バス	44	フェッч制御ユニット
12	コードROM	50	命令コード
13	RAM	10	並列処理フラグ
19	内部CPU	52	命令長コード
21	フェッчユニット	53	識別コード
22	デコード実行ユニット	54	オペランド用フィールド
23	インターフェースユニット	F R 0、F R 1、F R 2	フェッчレジスタ
31	専用データ処理ユニット	V R 0、V R 1	実行レジスタ
32	汎用データ処理ユニット	TR	テンポラルレジスタ

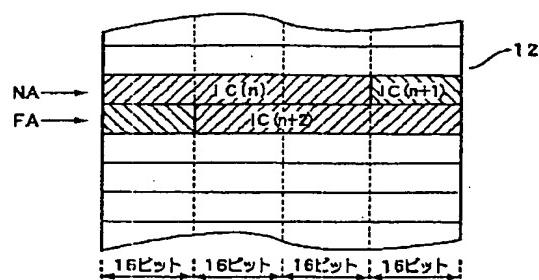
【図1】



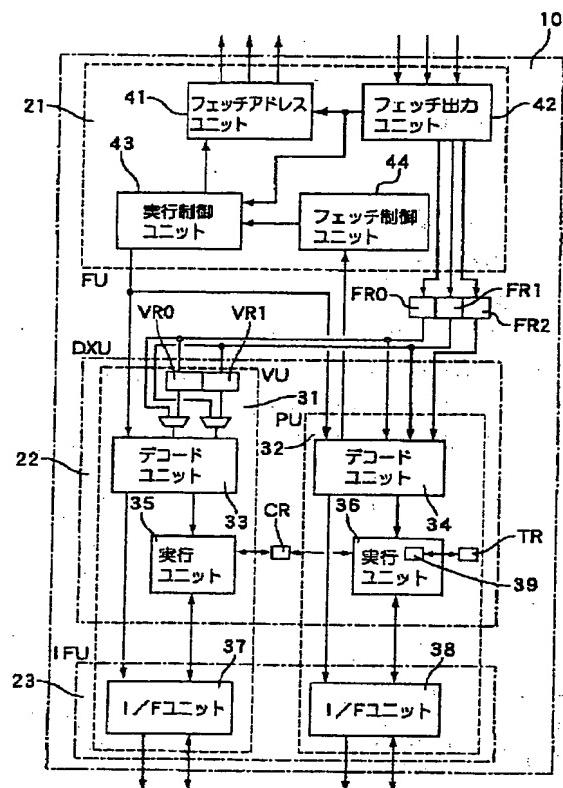
【図3】



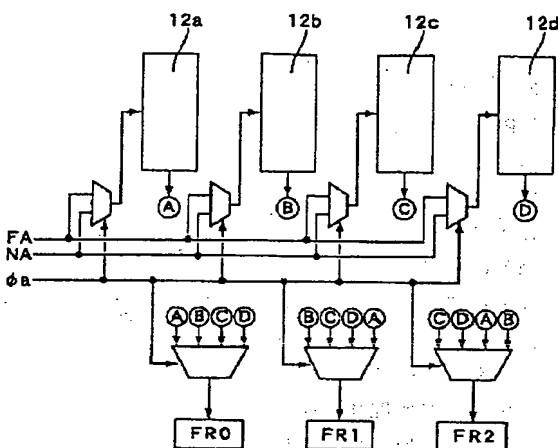
【図4】



【図2】



【図5】



【図6】

